# LFF MAKETPLACE PLATFORM SOLUTION ARCHITECTURE

**Module Development Plan for Land Fortune Marketplace (LFM)**

**Module 1: Core Identity & Membership Management System**

This is the foundational module that gates access to all others.

* **Sub-Module 1.1: User Onboarding & Registration**
  + **Details:** Multi-step form capturing entity type (Individual Farmer, Corporate Farm, Buyer, Input Supplier, etc.), personal/business details, tax IDs, bank account information, and geographic location (integrated with GIS for mapping).
  + **Function:** KYC (Know Your Customer) verification checks, document upload (scans of ID, business licenses). Assigns a unique LFM Member ID.
* **Sub-Module 1.2: Membership Tier Management**
  + **Details:** System to manage the three tiers:
    - **Registered Member:** Flags in DB granting access to basic marketplace features, loyalty programs, and member specials. No ZMX platform access.
    - **Clearing Member:** Additional flags granting full privileged access to the ZMX trading platform (futures, spot trading). Requires signed LFM Membership Agreement and link to a Margin Account.
    - **Non-Member:** Limited, read-only access to certain marketplace listings. No benefits.
    - **Loyalty Management:** We need a dashboard were buyers/customers can select; Product, Items, Consultancy Service, Items from the product and consultancy services catalogue.
  + **Function:** Automated permission sets based on membership level. Upgrade/downgrade workflows.
* **Sub-Module 1.3: Smart Card & Digital Profile Integration**
  + **Details:** A digital twin of the physical **Fortune Smart Card**. This sub-module stores and displays all data that would be on the card: Member ID, photo, transaction history, current **Fortune $** balance, linked Warehouse Receipts, loan status and input allocation history.
  + **Function:** QR code/barcode generation for the physical card. API endpoints for external systems (warehouses, banks) to validate the card and update transactions in real-time.

**Module 2: Financial Engine & Fort$ Management**

The economic circulatory system of the platform.

* **Sub-Module 2.1: Fort$ (Fortune Dollar) Ledger**
  + **Details:** A secure, internal ledger system that manages the creation, transaction, and redemption of the virtual currency. Each member has a Fort$ account balance.
  + **Function:** Records every Fort$ credit (e.g., from sale of goods) and debit (e.g., payment for inputs, services). Must be atomic (ensure transactions are all-or-nothing to prevent errors).
* **Sub-Module 2.2: Fiat Currency Gateway**
  + **Details:** Integration with banks and mobile money providers (e.g., EcoCash) via APIs. Allows members to top up their Fort$ accounts with real currency and cash out Fort$ into real currency.
  + **Function:** Processes deposits/withdrawals. Manages exchange rates between Fort$ and USD/ZWL. Handles transaction fees for LFF.
* **Sub-Module 2.3: Margin Account Management**
  + **Details:** Exclusive to **Clearing Members**. Interface with member banks to view and validate the balance in the member's designated Margin Account.
  + **Function:** The core of the "lock-in funds" requirement. Before a trade is placed, this system must receive confirmation (via API or secure upload) that sufficient funds are available and locked in the member's bank account.
* **Sub-Module 2.4: Payment Processing & Settlement**
  + **Details:** Orchestrates the end-of-trade financial settlement. When a trade is complete, it calculates the final amount, deducts LFF's fees, and initiates transfers.
  + **Function:** Instructs the Fort$ ledger to transfer Fort$ from buyer to seller. Or, via the banking gateway, instructs the bank to transfer fiat currency. Generates automated notifications to all parties.

**Module 3: Marketplace & Trading Hub (ZMX Platform Integration)**

The central nervous system for trading. This is the "ZMX Platform" mentioned throughout the document.

* **Sub-Module 3.1: Product Listing & Catalog Management**
  + **Details:** Allows sellers (and LFF itself) to list commodities (maize, cotton, livestock), inputs (seeds, fertilizer), and other sector goods (from construction to ICT). Fields must include: product type, grade, quantity, quality certifications, photos, and location (warehouse ID or farm location).
  + **Function:** Search and filter functionality for buyers. Inventory level management linked to the Warehouse Management System (WMS).
* **Sub-Module 3.2: Contract Management System**
  + **Details:** Digital creation, signing, and management of complex contracts.
    - **Forwards/Futures Contracts:** UI to set commodity, quantity, quality, price, and specific future delivery date.
    - **Spot Trade Contracts:** For immediate purchase and delivery.
  + **Function:** Automated alerts and notifications as contract maturity dates approach. Tracks contract fulfillment status.
* **Sub-Module 3.3: Auction & Bidding Engine**
  + **Details:** For commodities where price discovery is needed. Allows buyers to place bids, and sellers to accept the highest bid or set a reserve price.
  + **Function:** Real-time bid updating. Timer functionality. Automatic winner determination and contract generation.
* **Sub-Module 3.4: Requisition & Order Management**
  + **Details:** Manages the entire flow detailed in Figure 5. A "Purchase Requisition" is created by a buyer, triggering the fund lock-in process. Once funds are verified, it becomes an official "Order".
  + **Function:** Tracks the status of each requisition/order (e.g., Pending Funding, Approved, Awaiting Fulfillment, Completed). The core workflow engine for trades.

**Module 4: Warehouse Management System (WMS) & Receipt System**

Bridges the digital world with the physical inventory.

* **Sub-Module 4.1: Stock-In Processing**
  + **Details:** When a farmer delivers commodities to a CBP/Warehouse, the warehouse operator uses this interface. Records: Member ID (via Smart Card scan), commodity type, weight (from integrated scale), grade (from 3rd party inspector input), and moisture content.
  + **Function:** Creates a unique Lot ID for the received goods. This triggers the option to either pay the farmer immediately or generate a **Warehouse Receipt**.
* **Sub-Module 4.2: Digital Warehouse Receipt (WR) Generator**
  + **Details:** Generates a cryptographically secure, non-fungible digital receipt. The WR is a digital asset tied to a specific lot of commodities in a specific warehouse. It contains: WR ID, Lot ID, Commodity Details, Grade, Weight, Warehouse Location, Issue Date, and Owner ID (Farmer's LFM ID).
  + **Function:** The WR can be digitally transferred (e.g., sold on the marketplace, used as collateral). Its status must be trackable (Issued, Pledged, Redeemed).
* **Sub-Module 4.3: Stock-Out & Release Management**
  + **Details:** When a buyer comes to collect goods, the warehouse operator scans the buyer's Smart Card and the digital WR from the trade.
  + **Function:** Verifies the buyer is the legitimate owner of the WR. Records the release, including final weight from weigh-bridge. Updates inventory levels. Changes WR status to "Redeemed".
* **Sub-Module 4.4: Inventory Dashboard**
  + **Details:** Real-time view for warehouse managers and LFF administrators of stock levels across all designated warehouses and silos, broken down by commodity, grade, and age.
  + **Function:** Essential for logistics planning, reporting, and preventing stockouts.

**Module 5: Logistics Management System (Fortune Logistics)**

* **Sub-Module 5.1: Transportation Booking & Management**
  + **Details:** UI for members to request transport for goods from point A (farm) to point B (warehouse) or from warehouse to buyer. Includes details like commodity type, weight, volume, pickup/delivery locations and times.
  + **Function:** Automated pricing engine based on distance and commodity. Assigns shipments to available logistics partners via API integrations. Provides tracking numbers and real-time status updates (Picked Up, In Transit, Delivered).
* **Sub-Module 5.2: Customs & Permits Management (For International Trade)**
  + **Details:** A form-driven system to collect all necessary data for cross-border shipments. Integrates with official systems where possible, or provides printable, pre-filled forms.
  + **Function:** Manages Incoterms 2010 selection (FOB, CIF, etc.) and automatically calculates responsibilities based on the chosen term.

**Module 6: Data Management & Evaluation, Monitoring & Reporting (EMR)**

The "big data" brain of the operation.

* **Sub-Module 6.1: Geo-Tagging & GIS Integration**
  + **Details:** Integrates with **Area Extension Officers (AEOs)** tablets. Allows AEOs to drop pins on a map for each farm, attach photos of crops, and input progress reports.
  + **Function:** Creates a visual, data-rich map of all registered farms, enabling monitoring of planting, growth, and harvest progress. Critical for crop forecasting and insurance assessment.
* **Sub-Module 6.2: Analytics & Reporting Dashboard**
  + **Details:** A powerful dashboard for LFF, financiers, and large buyers. Provides insights on: commodity price trends, regional production volumes, member trading activity, platform revenue, and logistics performance.
  + **Function:** Customizable reports. Data visualizations (charts, graphs). Ability to export data. This is a key sellable product to stakeholders.
* **Sub-Module 6.3: API Gateway**
  + **Details:** A secure set of APIs (Application Programming Interfaces) to allow controlled access to the DMS for partners: Banks, ZMX, Insurance companies, 3rd party inspectors.
  + **Function:** A bank's system can query a member's balance or WR ownership. An insurer can pull field data to assess a claim. This interoperability is critical.

**Module 7: Value-Added Services Portal**

* **Sub-Module 7.1: Insurance Marketplace**
  + **Details:** A curated portal within LFM where members can browse and purchase the listed insurance products (Agro, Health, Funeral, etc.) from partner providers.
  + **Function:** Secure checkout using Fort$ or linked payment methods. Application forms and digital policy issuance.
* **Sub-Module 7.2: Inputs Acquisition & Credit Facility**
  + **Details:** Digital catalog of inputs from suppliers. Farmers can apply for inputs on credit, which creates a loan obligation.
  + **Function:** Manages the entire workflow from Figure 2: application, eligibility check, approval, electronic notification to warehouse, and ultimately deducting the cost from the farmer's eventual sales proceeds.

**Module 8: Notification & Communication Engine**

* **Details:** A central system that triggers all automated communications.
* **Function:** Sends SMS, email, and mobile app push notifications for: trade confirmations, payment receipts, contract maturity alerts, warehouse receipt issuance, AEO visit schedules, and platform announcements. This is not a UI module but a critical backend service.

**Module 9: Administration & Compliance Panel**

* **Details:** A super-user panel for LFF staff to manage the entire platform.
* **Function:** User management, role-based access control, fee configuration, dispute resolution tools, audit logs, membership suspension/termination, and system health monitoring. Includes tools to enforce the detailed **Trading Rules and Regulations** (e.g., automatically blocking trades for members with overdue fees).  
    
    
  **CORE WORKFLOWS FOR LFF MARKETPLACE APP**

|  |  |  |
| --- | --- | --- |
| Work Item Type | Description | Key Microservices Involved |
| **Order** | Sale of commodities, livestock, or expert services. | lff-ecommerce-api, lff-payments-integration-api, lff-logistics-api, lff-warehouse-api, lff-commodity-livestock-api, lff-odoo-invoice-api |
| **Permit / Collateral Request** | Regulatory clearance or collateral pledge for certain commodities/livestock. | lff-perm-clearing-collateral-api, lff-user-compliance-api |
| **Delivery / Shipment** | Movement of goods or livestock to buyer. | lff-logistics-api, lff-tag-trace-api, lff-loconav-integration-api |
| **Insurance Policy / Claim** | Coverage of shipments or livestock. | lff-insurance-broking-api |
| **Customer Account Lifecycle** | Registration, KYC, roles, offboarding. | lff-user-compliance-api, lff-customer-management-api |
| **Reward / Incentive Item** | Loyalty points, discount coupons. | lff-customer-rewards-api, lff-smart-accounts-api |
| **Payment Transaction** | Multi-currency payments (ZiG/USD), refunds. | lff-payments-integration-api, lff-smart-accounts-api |

|  |  |
| --- | --- |
| Status | Definition |
| **Draft** | Work item created but not submitted; editable by originator. |
| **Pending Verification** | Awaiting compliance/KYC/permit or collateral checks. |
| **Pending Payment** | Awaiting full or partial payment confirmation. |
| **Payment Failed** | Payment attempt unsuccessful; retry allowed. |
| **Payment Confirmed** | Funds received and verified. |
| **Awaiting Fulfilment** | Ready for warehouse pick/packing or scheduling of livestock delivery. |
| **In Transit** | Logistics partner has collected; en route to destination. |
| **Delivered / Completed** | Item received by buyer and acknowledged. |
| **Partially Fulfilled** | Only part of order shipped/received (e.g., split shipments). |
| **Cancelled** | Cancelled by buyer or system before fulfilment. |
| **Rejected** | Compliance/permit failed; cannot continue. |
| **Returned / Disputed** | Buyer initiated return or dispute. |
| **Closed** | Financials reconciled; no further action allowed. |
| **Expired** | Timed out (e.g., permit not obtained, payment window lapsed). |

## **Typical Status Transitions**

Below is a **general state diagram** (per work item) that all microservices can reference through lff-workflow-api  
  
Draft  
├─> Pending Verification ──> Pending Payment ──> Payment Confirmed  
│ │  
│ ├─> Payment Failed ──> Pending Payment (retry)  
│  
├─> Cancelled  
└─> Expired  
Payment Confirmed ──> Awaiting Fulfilment ──> In Transit ──> Delivered/Completed  
│ │  
├─> Partially Fulfilled  
└─> Cancelled (if still in warehouse, not in transit)  
Delivered/Completed ──> Returned/Disputed ──> Closed  
Cancelled / Rejected / Expired ──> Closed

|  |  |
| --- | --- |
| Invalid Transition | Rationale |
| **Draft → In Transit** | Must pass verification & payment first. |
| **Cancelled → Pending Payment** | Once cancelled, financial obligations are void. |
| **Delivered/Completed → Pending Payment** | Delivery can’t revert to unpaid. |
| **Closed → Any Other Status** | Closed is a terminal state. |
| **Rejected → Pending Verification** | Rejection requires a new request, not a re-check. |
| **Payment Failed → In Transit** | Goods can’t ship without confirmed payment. |
| **Expired → Pending Payment** | Expired items must be recreated, not reactivated. |
| **Partially Fulfilled → Draft** | Can only move to Awaiting Fulfilment (for remaining quantity) or Delivered. |
| **Returned/Disputed → In Transit** | Returns/disputes are post-delivery only. |

The lff-workflow-api should **centrally enforce** these transition rules with role-based checks from lff-user-compliance-api.  
  
**Microservice Alignment**

|  |  |
| --- | --- |
| Microservice | Workflow Responsibilities |
| lff-workflow-api | Master state machine; publishes state-change events. |
| lff-notification-api | Sends SMS/Email/WhatsApp/Push for each significant state change. |
| lff-user-compliance-api | Validates KYC/AML before allowing “Pending Verification → Pending Payment”. |
| lff-logistics-api **/** lff-tag-trace-api **/** lff-loconav-integration-api | Enforce “Awaiting Fulfilment → In Transit → Delivered”. |
| lff-payments-integration-api **/** lff-smart-accounts-api | Control all payment-related transitions and reversals. |
| lff-customer-rewards-api | Triggers reward accrual at “Delivered/Completed”. |
| lff-odoo-invoice-api | Auto-issues invoices at “Payment Confirmed”. |
| lff-perm-clearing-collateral-api | Blocks order movement until permits/collateral verified. |

## **Zimbabwe-Specific Considerations**

* **Currencies**: ZiG & USD; support split payments and official exchange rates.
* **Regulatory Holds**: Certain agricultural exports need government permits. Add a **“Regulatory Hold”** sub-status inside **Pending Verification**.
* **Insurance Compliance**: Some livestock shipments may require insurance before **In Transit**.

### **Implementation Tips**

* **Event-Driven**: Use Kafka topics for order.status.changed events consumed by all dependent services.
* **Immutable Logs**: lff-workflow-api must keep an **append-only audit trail** for ZIMRA/Reserve Bank compliance.
* **Role-Based Guards**: Integrate with lff-user-compliance-api to restrict who can execute transitions like “Cancel after Payment”.

**User Management Workflow**

|  |  |
| --- | --- |
| Status | Definition |
| **Pending Registration** | User submitted sign-up but email/phone not yet verified. |
| **Verification Pending** | Awaiting KYC/AML and ID checks (ZIMRA or national ID). |
| **Active** | Fully verified and allowed to transact. |
| **Suspended** | Temporarily disabled (fraud, inactivity, non-compliance). |
| **Deactivated** | User voluntarily closed account. |
| **Deleted** | Soft delete (GDPR/POPIA compliance). |

**Order Management Workflow**

|  |  |
| --- | --- |
| Status | Definition |
| **Draft** | Cart saved, not submitted. |
| **Pending Verification** | Waiting for permit/collateral checks (e.g., livestock movement permit). |
| **Pending Payment** | Order verified, payment window open. |
| **Payment Failed** | Payment attempt unsuccessful. |
| **Payment Confirmed** | Funds received & matched. |
| **Awaiting Fulfilment** | Ready for warehouse pick/packing or expert service scheduling. |
| **In Transit / Service In Progress** | Shipment moving or expert service being performed. |
| **Delivered / Completed** | Buyer received goods or service finished. |
| **Partially Fulfilled** | Split shipments or partial service delivered. |
| **Cancelled** | Cancelled by buyer or system before fulfilment. |
| **Rejected** | Compliance or stock failure. |
| **Returned / Disputed** | Buyer initiated return or dispute. |
| **Closed** | Financials and any returns settled. |
| **Expired** | Payment or permit window lapsed. |

Draft ──> Pending Verification ──> Pending Payment ──> Payment Confirmed  
Payment Confirmed ──> Awaiting Fulfilment ──> In Transit ──> Delivered/Completed ──> Closed  
  
**Payment Management Workflow**

**Work Item Type:** Payment Transaction (ZiG / USD / multi-method)

|  |  |
| --- | --- |
| Status | Definition |
| **Initiated** | Payment request created. |
| **Pending Authorization** | Waiting for OTP, 3-D Secure, or bank approval. |
| **Authorized** | Bank/processor approved, funds not yet settled. |
| **Captured / Settled** | Funds received and posted. |
| **Failed** | Authorization or capture failed. |
| **Refund Pending** | Refund requested. |
| **Refunded** | Funds returned to payer. |
| **Chargeback / Dispute** | Buyer raised chargeback. |
| **Closed** | Fully reconciled. |

**Violations**

* Closed → Any
* Failed → Authorized (cannot re-authorize; must create new transaction)
* Refunded → Captured (no “un-refund”)

**Microservices:**  
lff-payments-integration-api, lff-smart-accounts-api, lff-notification-api, lff-workflow-api.  
  
Cross-Domain Guardrails

* **Order–Payment Link:**  
  Order cannot advance to “Awaiting Fulfilment” unless at least one related Payment is “Captured/Settled”.
* **User–Order Link:**  
  Orders tied to Suspended/Deactivated/Deleted users auto-freeze in “On Hold” sub-status until user is Active again.
* **Compliance Hold:**  
  If lff-user-compliance-api flags a KYC breach, transition to a **Regulatory Hold** status overrides all progress.

## **Integration Touchpoints**

* **Event-Driven:** All services publish/consume status.changed events via Kafka (lff-workflow-api as source of truth).
* **Notifications:** lff-notification-api sends SMS/Email/WhatsApp/Push on each critical transition.
* **Audit & Immutability:** Append-only history maintained for ZIMRA and Reserve Bank of Zimbabwe compliance.

|  |  |  |
| --- | --- | --- |
| Domain | Key End States | Terminal? |
| **User** | Deleted, Deactivated | Yes |
| **Order** | Closed, Cancelled, Rejected | Yes |
| **Payment** | Closed, Failed, Refunded | Yes |

## **Insurance Broking workflow**

Below is a **dedicated Insurance Broking workflow blueprint** that fits naturally into your Zimbabwe-based agricultural e-commerce ecosystem and aligns with your other microservices and workflow APIs.  
Work Item Type

**Insurance Policy / Claim**

Covers:

* **Shipment Cover** – protects commodities/livestock while in transit.
* **Livestock Cover** – mortality, theft, disease.
* **Expert Services Cover** – liability for agronomy/vet consultants.

Key microservices:  
lff-insurance-broking-api, lff-payments-integration-api, lff-smart-accounts-api, lff-user-compliance-api, lff-notification-api, lff-workflow-api.  
  
**Insurance Policy Lifecycle & Status Definitions**

|  |  |
| --- | --- |
| Status | Definition |
| **Quote Requested** | Customer (or Order workflow) requests a policy quote. |
| **Quote Provided** | Underwriter returns pricing & terms. |
| **Quote Accepted** | Customer accepts terms; pending compliance & payment. |
| **Pending Underwriting** | Insurer performing detailed risk/permit checks. |
| **Pending Payment** | Awaiting premium payment confirmation (ZiG/USD). |
| **Payment Failed** | Attempted payment did not succeed. |
| **Policy Active** | Premium received; coverage in force. |
| **Policy Lapsed** | Renewal or instalment premium missed. |
| **Policy Cancelled** | Cancelled by customer or broker before term end. |
| **Claim Filed** | Customer submitted claim. |
| **Claim Under Review** | Adjuster investigating claim. |
| **Claim Approved** | Claim validated, ready for payout. |
| **Claim Rejected** | Claim denied (fraud or exclusion). |
| **Claim Paid** | Funds disbursed to claimant. |
| **Closed** | Policy term ended or claim fully resolved; no further action. |

## **Allowed Transitions | Insurance Broking**

### **Policy Creation Flow**

Quote Requested → Quote Provided → Quote Accepted → Pending Underwriting  
→ Pending Payment → Policy Active  
Policy Active → Policy Lapsed → Closed  
Policy Active → Policy Cancelled → Closed  
  
Policy Active → Claim Filed → Claim Under Review  
Claim Under Review → Claim Approved → Claim Paid → Closed  
Claim Under Review → Claim Rejected → Closed  
  
  
**Transition Violations (Disallowed Paths) | Insurance Broking**

|  |  |
| --- | --- |
| From → To | Why Disallowed |
| **Quote Requested → Policy Active** | Must have quote, acceptance, and payment. |
| **Policy Cancelled → Policy Active** | Cancelled is final; needs new policy. |
| **Closed → Any** | Terminal state. |
| **Claim Rejected → Claim Under Review** | Requires new claim, not reopen. |
| **Payment Failed → Policy Active** | Must retry payment first. |
| **Policy Lapsed → Pending Payment** | Renewal creates a new cycle, not revert. |

## **Integration Touchpoints | Insurance broking**

|  |  |
| --- | --- |
| Step | Microservice Actions |
| Quote generation | lff-insurance-broking-api calls external underwriter APIs. |
| Customer acceptance | lff-user-compliance-api checks KYC & agricultural permit obligations. |
| Premium payment | lff-payments-integration-api & lff-smart-accounts-api confirm funds. |
| Policy issuance & storage | Document stored via lff-odoo-invoice-api or Paperless-NGX (if used). |
| Status change notifications | lff-notification-api sends email/SMS/WhatsApp/Push to insured parties. |
| Claims | Adjusters update statuses through lff-insurance-broking-api; payouts triggered in lff-payments-integration-api. |

## **Zimbabwe-Specific Considerations**

* **Regulatory** – Policies must comply with **IPEC** (Insurance and Pensions Commission) rules; “Pending Underwriting” may include an **IPEC Approval** sub-status.
* **Currency** – Premiums and payouts in **ZiG or USD** with official RBZ exchange rates.
* **Livestock Identification** – Link lff-tag-trace-api data to claim validation for theft or loss.

|  |  |
| --- | --- |
| Domain | Terminal Statuses |
| **Insurance Policy** | Closed, Policy Cancelled, Policy Lapsed |
| **Claim** | Claim Paid, Claim Rejected (both feed into Closed) |

===========**LFF Marketplace Project ||| FINAL Microservice API List**==============  
lff-notification-api (SMS, Email, Whatsapp, Push/ in-App)  
lff-user-compliance-api (similar to PASA PDMS)  
lff-logistics-api  
lff-workflow-api  
lff-customer-management-api  
lff-perm-clearing-collateral-api  
lff-trading-api  
lff-insurance-broking-api  
lff-evaluation-monitoring-api  
lff-odoo-invoice-api  
lff-warehouse-api  
lff-ecommerce-api  
lff-commodity-livestock-api  
lff-tag-trace-api  
lff-payments-integration-api  
lff-ecommerce-api  
lff-smart-accounts-api  
lff-customer-rewards-api  
lff-loconav-integration-api  
lff-zmex-integration-api  
lff-kube-cluster-configs  
lff-mobile-app  
lff-web-app

lff-commons

**LFF Marketplace Key Modules Notes**  
==================================  
1.Ecommerce  
-e-Store  
-Abbatoir (selling of carcasses for various animals etc)  
2.Rename Delivery to Logistics  
3.LiveStock Tagging & Tracking  
4.Invoicing [Integration with ODOO ERP] [Reconciliation]  
5.Warehouse  
6.Insurance Broking []  
-Funeral  
-Health  
-Homeowners  
-Commercial  
-Sector Specific  
-Umbrella  
7.Trading  
-Agricultral/Farming  
-Construction  
-Energy  
-Engineering  
-Equipment  
-Fuels  
-Food & Beverage  
-ICT  
-Irrigation  
-Manufacturing  
-Mining  
-Power  
-Service Sectors  
-Water  
8.LFF Customer Management  
9.LFF Smart Accounts  
10.LFF Customer Rewards  
-Loyalty & Rewards  
-Incentives, Discounts, Special Offers  
11.Permits, Clearing & Collateral  
12.Payments [Petros Muchato | Reconciliations]  
13.Evaluation & Monitoring (Harvesting, Agric EXT workers etc)  
14.User Management (Pending, Active, Revoke=[Misconduct, 3 Login Attempts, Suspended, User Roles)  
15.Audit